yang_seonhyeHW14

Question 1

We know that $\beta = (X^TX)X^{-1}y$ and X^TX calculates the correlation.

$$\hat{\beta}_r = (X^TX + \lambda I)^{-1}X^Ty$$

$\hat{\beta} = \hat{\beta}_r$ only if $\lambda = 0$

$$\hat{\beta}_r = (X^TX + \lambda I)^{-1}X^Ty = \hat{\beta}_r = ((X^TX)^{-1}\lambda + I)^{-1}$$

$$(X^TX)X^{-1}y = ((X^TX)^{-1}\lambda + I)^{-1\beta}$$

$$Var(aX) = aVar(X)a$$

$$Var(\hat{\beta}_r) = \sigma^2(X^TX + \lambda I)X^TX(X^TX + \lambda I)$$

Question 2

From the previous question, we have $Var(\hat{\beta}_r) = \sigma^2(X^TX + \lambda I)X^TX(X^TX + \lambda I)$ and $Var(\hat{\beta}_r) = \sigma^2 I$ where $(X^TX + \lambda I)X^TX(X^TX + \lambda I) = I$

Therefore, $\sigma^2 I = \sigma^2(X^TX + \lambda I)X^TX(X^TX + \lambda I)$

Question 3

We know that $E(\hat{\epsilon}) = E(y - \hat{y_r}), \hat{\beta}_r = ((X^TX)^{-1}\lambda + I)^{-1}$:

$$E(X\hat{\beta} - X\hat{\beta_r}) = XE(\hat{\beta} - ((X^TX)^{-1}\lambda + I)^{-1}\beta)$$

$$X(-((X^TX)^{-1}\lambda + I)^{-1})\beta$$

where $\beta \neq 0$

Question 4

```
library(alr4)
```

```
## Loading required package: car
```

```
## Loading required package: carData
```

```
## Loading required package: effects
```

```
## lattice theme set by effectsTheme()
## See ?effectsTheme for details.
```

```
library(MASS)
library(data.table)
attach(MinnWater, warn.conflicts = FALSE)
data(MinnWater)
apply(MinnWater,2,sd)
```

```
##         year       allUse      muniUse       irrUse     agPrecip
## 7.071068e+00 3.274575e+01 1.313644e+01 2.122745e+01 2.597403e+00
##   muniPrecip     statePop      muniPop
## 4.399553e+00 3.307580e+05 2.566059e+05
```

```
apply(MinnWater,2,mean)
```

```
##         year       allUse      muniUse       irrUse     agPrecip
## 1.999500e+03 2.348042e+02 1.233000e+02 6.292917e+01 1.142917e+01
##   muniPrecip     statePop      muniPop
## 1.992917e+01 4.862009e+06 2.951255e+06
```

```
MinnWater_st = as.data.frame(scale(MinnWater))
apply(MinnWater_st,2,sd)
```

```
##       year     allUse    muniUse     irrUse   agPrecip muniPrecip
##          1          1          1          1          1          1
##   statePop    muniPop
##          1          1
```

```
apply(MinnWater_st,2,mean)
```

```
##          year        allUse       muniUse        irrUse      agPrecip
##  0.000000e+00 -1.879871e-16  3.269954e-16 -4.227937e-17 -1.335150e-16
##    muniPrecip      statePop       muniPop
## -5.666312e-17  9.362538e-16 -6.878179e-16
```

```
model = lm.ridge(muniUse~.,data=MinnWater_st,lambda=seq(0,0.1,0.001))
select(model)
```

```
## modified HKB estimator is 0.002658801
## modified L-W estimator is 0.0200445
## smallest value of GCV  at 0.004
```

Question 5

```
fit1 = lm(muniUse~.,data = MinnWater_st)
summary(fit1)
```

```
##
## Call:
## lm(formula = muniUse ~ ., data = MinnWater_st)
##
## Residuals:
##       Min        1Q    Median        3Q       Max
## -0.105686 -0.033746  0.001576  0.036447  0.096089
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) -1.783e-15  1.264e-02   0.000  1.00000
## year        -2.026e-01  3.248e-01  -0.624  0.54167
## allUse       1.722e+00  1.780e-01   9.677 4.33e-08 ***
## irrUse      -1.056e+00  1.403e-01  -7.528 1.21e-06 ***
## agPrecip     3.210e-02  3.088e-02   1.040  0.31398
## muniPrecip  -8.343e-02  2.673e-02  -3.122  0.00658 **
## statePop     1.531e+00  5.633e-01   2.719  0.01517 *
## muniPop     -1.021e+00  7.572e-01  -1.349  0.19621
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.0619 on 16 degrees of freedom
## Multiple R-squared:  0.9973, Adjusted R-squared:  0.9962
## F-statistic: 855.2 on 7 and 16 DF,  p-value: < 2.2e-16
```

```
X = as.matrix(MinnWater_st[,-3])
ols_se = sqrt(diag(sigma(fit1)^2*solve(t(X)%*%X)))
ols_se
```

```
##       year     allUse     irrUse   agPrecip muniPrecip   statePop
## 0.32482380 0.17798310 0.14027649 0.03087536 0.02672613 0.56326376
##    muniPop
## 0.75717031
```

```
#Ridge Regression Standard Error
ridge_se <- function(lambdaval, covs, mods) {
  lambdaMat <- diag(rep(lambdaval,7),7,7)
  tmp <- solve((t(covs)%*%covs+lambdaMat))
  se <- sqrt(diag(mods^2*tmp%*%t(covs)%*%covs%*%tmp))
  return(se)
}
r_se <- ridge_se(0.004, X, 0.0619)
r_se
```

```
##       year     allUse     irrUse   agPrecip muniPrecip   statePop
## 0.23028780 0.14616732 0.12168730 0.03051301 0.02350918 0.31081817
##    muniPop
## 0.38707818
```

```
r_se - ols_se
```

```
##          year        allUse        irrUse      agPrecip    muniPrecip
## -0.0945359912 -0.0318157855 -0.0185891868 -0.0003623489 -0.0032169525
##      statePop       muniPop
## -0.2524455911 -0.3700921291
```

From our result, we can see that the ridge model had lower coefficient standard errors for every coefficient, indicating that it is the better model